Hindawi
Security and Communication Networks
Volume 2019, Article ID 3745619, 13 pages
https://doi.org/10.1155/2019/3745619

Research Article

Modelling the Spread of Botnet Malware in IoT-Based Wireless Sensor Networks

Dilara Acarali,[1] **Muttukrishnan Rajarajan**,[1] **Nikos Komninos**,[1] **and B. B. Zarpelão**[2]

[1]*School of Mathematics, Computer Science and Engineering, City, University of London, UK*
[2]*Computer Science Department, State University of Londrina, Brazil*

Correspondence should be addressed to B. B. Zarpelão; brunozarpelao@uel.br

Received 5 December 2018; Revised 9 January 2019; Accepted 15 January 2019; Published 3 February 2019

Academic Editor: Angel M. Del Rey

The propagation approach of a botnet largely dictates its formation, establishing a foundation of bots for future exploitation. The chosen propagation method determines the attack surface and, consequently, the degree of network penetration, as well as the overall size and the eventual attack potency. It is therefore essential to understand propagation behaviours and influential factors in order to better secure vulnerable systems. Whilst botnet propagation is generally well studied, newer technologies like IoT have unique characteristics which are yet to be thoroughly explored. In this paper, we apply the principles of epidemic modelling to IoT networks consisting of wireless sensor nodes. We build `IoT-SIS`, a novel propagation model which considers the impact of IoT-specific characteristics like limited processing power, energy restrictions, and node density on the formation of a botnet. Focusing on worm-based propagation, this model is used to explore the dynamics of spread using numerical simulations and the Monte Carlo method to discuss the real-life implications of our findings.

1. Introduction

IoT networks are increasingly becoming a feature of our digital landscape. These networks consist of devices with sensing capabilities designed to collect data on the environment, which is then forwarded via sink nodes to be processed. This data can then be used to learn about customers, improve efficiency, or supplement services. IoT sensor networks are characteristically different to conventional networks. Sensor devices are low powered and often use batteries as their primary source of energy. Therefore, energy efficiency is a priority. These power restrictions mean that devices have limited processing capabilities, which often results in poor security. Sensor networks also tend to be dense. This is due to the requirements of data collection (i.e., the types of data desired and the coverage needed). These unique characteristics have an influence on the propagation of malware and the development of potential botnet threats.

In 2016-17, the Mirai botnet was able to gain traction and, as a result, grabbed public attention with a series of high-profile, large-scale DDoS attacks [1]. Using a relatively simple propagation approach, Mirai was able to quietly spread to many devices, estimated to be around 600,000 at its peak [2]. This provided a large pool of bots to draw from, resulting in DDoS attacks with a huge force of 620 Gbps against a security blog [1, 3] and 1.1 Tbps against a French Internet provider [1], both in 2016. The events around Mirai demonstrate the prominent threat of botnets in the IoT space. To make matters worse, Mirai's source code was made public, and multiple spinoffs and copycats have already been reported including Persirai [1], BrickerBot [1], and HideNSeek [4].

Propagation is typically a difficult process to detect and to observe. This is because different vulnerabilities emerge across different technologies, various services or functions can serve as propagation vectors, and we collectively lack experience with widespread implementation of new technologies like IoT or IPv6. Consequently, the initial propagation process tends to be revealed in hindsight, only after an existing infection has been identified. Therefore, we use modelling approaches based on our understanding of the technology and experience with historic attacks to predict propagation dynamics and to explore influential factors.

We apply this approach by building `IoT-SIS`, a model of botnet propagation through IoT-based wireless sensor

networks, focusing on the unique characteristics of IoT that differentiates it from other types of network. `IoT-SIS` itself is based on epidemiological concepts and uses the *SIS* (Susceptible, Infected, Susceptible) paradigm as its foundation. In conducting this research, we hope to better understand how botmasters may approach IoT networks and what factors are most influential from a defensive perspective. This includes a consideration of the botmasters requirements, such as the need to balance the acquisition of new bots with the need to sustain the existing bot population. This topic is not currently well studied, and we hope to address this with the following contributions:

(i) A novel *SIS*-based model, called `IoT-SIS`, of IoT-based worm propagation and botnet formation

(ii) An in-depth exploration of the relationships between various factors (such as bot activity, node hardware, and deployment scheme) via simulations

(iii) An analysis of the model using the Monte-Carlo simulation method

The paper is organised as follows: Section 2 provides a background on epidemiological modelling, IoT, and known IoT-based botnets. Section 3 defines the model and describes the parameters, with rationale for each choice. In Section 4, we outline our simulation setup and present our results. Section 5 discusses our findings and makes suggestions for defence and future work. Examples of related work are presented in Section 6, and we conclude in Section 7.

2. Background

2.1. Epidemic Modelling. The medical field of epidemiology is the study of disease incidence in populations, used to analyse spread dynamics and to measure potential immunisation strategies. Based on the work of Kermack and McKendrick [5], epidemic principles are used to mathematically model the outbreak of infectious diseases where scientific experimentation is not feasible or ethical [6], allowing researchers to predict possible impact factors in transmission dynamics, which then feeds into the development of public health policies [6]. Epidemic modelling was introduced to cybersecurity by Kephart and White [7] in their study of computer viruses, where they used populations of computer systems, substituted malware for diseases, and based contact on network communication graphs. This has developed further to apply to various types of malware, including botnets where such models allow us to consider the factors impacting the size of bot populations.

Epidemic models consist of states or compartments, coupled with some transition conditions that determine when a node moves from one state to another. These models are sometimes referred to as compartmental models, as the total population is divided amongst a number of compartments based on their current status [6]. States (or compartments) are designed to abstractly describe the current role played by nodes, encapsulating any behaviours and characteristics that may be associated with that role. The number of states reflects the number of possible roles that nodes may take. The system is then measured by considering rates of change and calculating the number of nodes within each compartment over time. In compartmental models, transitions are typically defined as a system of differential equations, commonly featuring elements such as the rates of contact, infection, recovery, births, and deaths.

The most basic epidemic model is *SI*, consisting of the Susceptible (S) and Infected (I) states. 'Susceptible' describes a vulnerable individual who has not yet been infected, whilst 'Infected' tends to denote an individual who is both a carrier and a propagator of the pathogen. Nodes would then transition from S to I at the rate of infection. The *SIR* model adds the Recovered (R) state to represent individuals who have been healed and subsequently gained immunity. Nodes will transition from I to R at the rate of recovery. An alternative is *SIS*, where recovered nodes do not gain immunity but instead return to their previous susceptible status. Finally, the *SEIR* model adds the Exposed (E) state to denote infected individuals who are either asymptomatic or not able to pass on the pathogen until they transition into the I state. This is used where the incubation period of diseases needs to be considered. Basic versions of the epidemic models are generally deterministic but may include probabilistic elements. Stochastic versions of these models tend to use Markovian Processes or stochastic differentials [6].

In this work, the model incorporates probabilistic elements to more accurately represent the likelihoods of contact and infection. Additionally, we based our approach on the *SIS* format. This is because IoT malware often runs in the RAM and is not persistent, meaning that rebooting can clean the sensor of an infection. However, the node does not gain immunity. Meanwhile, devices which are recovered via patching are likely to fall victim to the same malware again because bots frequently receive updates containing new exploits. Hence, in both scenarios, it is realistic to consider recovered nodes susceptible to reinfection rather than permanently immune.

2.2. IoT Sensor Networks. IoT sensor networks consist of wireless sensor nodes, which are small devices equipped with the ability to sense the environment and to perform small computations [8]. Devices form a wireless sensor network (WSN) to collaboratively sense and respond to the environment [8] and also to communicate with IoT-enabled devices like routers, allowing access to the wider infrastructure for data retrieval and processing. WSNs are made up of sensor nodes and sink nodes. Sink nodes act as a hub for data collection and as a gateway for the WSN [9]. Users may observe the IoT-based WSN directly, via a local IP-based network, or remotely over the Internet and can send commands via sink nodes [9]. The defining characteristics of WSNs are summarised in Table 1.

Sensor nodes consist of 4 base parts: the power unit, the sensor, the processor, and the radio [9]. The sensor measures environmental variables, whilst the radio handles communication. The processor arranges tasks and deals with the conversion of data into signals for transmission, whilst the power unit consists of the node's battery pack

Table 1: Key characteristics of IoT sensors in WSNs.

Characteristic	Description
Restricted energy	Nodes must conserve their batteries.
Restricted processing	Limited capacity due to low power.
Dense deployment [9]	High density for better coverage.
Application-specific [9]	Designed for particular sensing tasks.
Many-to-one traffic	Many nodes forward data to 1 sink.

[9]. The processor also manages sleep cycles, used by nodes to conserve power. Typically, the most energy-consuming function is the exchange of data, with the degree of energy required increasing exponentially the further the data needs to travel [9]. Hence, node density and deployment patterns must be considered carefully.

The IoT stack is structured similarly to the TCP/IP stack with 5 horizontal layers defining end-to-end communication from the physical medium (layer 1) up to the application (layer 5). It also includes additional vertical 'planes' [9], representing processes which must be managed at each layer. These are (a) Power (i.e., the sharing of power between node functions), (b) Mobility (i.e., the tracking of nodes), and (c) Tasks, (i.e., communication, message detection, and sensing activities). Protocols at each layer must address the 3 vertical processes [8].

For the botmaster, these processes may highlight areas of vulnerability. For example: (a) high-power consumption on infected nodes can result in node death, (b) node mobility can be exploited to join WSNs in Sybil-style attacks, and (c) task schedules may be manipulated to steal information. For captured nodes, these processes also need to be considered as part of the botnet's maintenance. In the Power plane, the botmaster must limit bots' activity levels to avoid power depletion as this would hurt their propagation gains. In the Tasks plane, this may involve the cancellation of scheduled tasks or the disabling of services. Meanwhile, in the Mobility plane, GPS tracking on mobile nodes may reveal new targets.

There are currently several IoT communication standards available; in this work, we focus on 6LoWPAN (IPv6 over Low-Power WPAN) and RPL (Routing Protocol for Low-Power and Lossy Networks) [10] which are designed specifically for LLNs (Low-Power and Lossy Networks) (i.e., constrained networks) such as IoT-based WSNs [10]. Based on the IEEE 802.15.4 network standard, they provide IPv6-based routing functionality [8] to connect sensor networks to IP networks. The migration to IPv6 is necessary due to the massively increased number of Internet-connected devices in need of unique identifiers [8]. 6LoWPAN runs on low-energy sensor devices and adds an interface layer to allow compatibility between the IP-based routing and the lower IEEE 802.1.5.4-based layers [8].

RPL arranges nodes into DoDAGs (Destination-orientated Directed Acyclic Graphs) to enable routing. In these graphs, nodes form parent-child relationships, anchored by a root node which is the edge router connecting the WSN to the IP network. Parent-child relationships are based on an OF (Objective Function), a user-defined metric for route optimisation [11]. The neighbour representing the optimum path towards the root is hence selected as the preferred parent. Consequently, IEEE 802.15.4 also allows nearby nodes to have P2P-based communication.

2.3. IoT Botnets. Mirai caused widespread disruption during 2016 and 2017 with a series of large-scale DDoS attacks. According to [2], 65,000 devices were infected in 20 hours, and the botnet achieved a peak size of 600,000 nodes [2]. Mirai uses worm-based propagation, which is characterised by periods of scanning for vulnerable devices, reportedly targeting IoT-enabled cameras, routers, printers, and video recorders during its "rapid scanning phase" [1, 2]. The malware sends TCP SYN messages to random IPv4 addresses on ports 23 and 2323. For successful connections, it then tries to access the device using a dictionary attack based on 62 commonly used default logins credentials. If successful, the logins and the device IP are recorded on a server which then triggers a loader to download the malware on to the target [2]. Mirai sometimes kills existing processes [2], which may be a defence against other malware or a method of preserving energy. Antonakakis et al. [2] noted that Mirai's scanning rates and subsequent infection rates were lower than that of other known worms such as Code Red or Blaster, suggesting that limited device capacity may be the cause.

After Mirai's source code was made public, several derivatives have been reported [1]. One such derivative is the worm-based bot Persirai. Discovered in April 2017, it reportedly uses port 81 and known exploits to gain access to password files on IoT webcams before targeting routers via UPnP exploits [1]. BrickerBot, also identified in April 2017, uses default SSH logins and known exploits to permeate IoT devices, before corrupting firmware and generally debasing devices to make then unusable [1]. More recently, BitDefender reported a botnet called HideNSeek, detected in January 2018 [4]. Suggested to be in its expansion phase, it propagates by randomly scanning the IP space with SYN connections on ports 23, 2323, 80, and 8080 [4], and when a connection is established, attempts a Mirai-style dictionary attack.

The IoT botnets observed to date find victims by scanning the network, target similar open ports, and use exploits or weak credentials for penetration. Overall, they are characterised by their simplicity, using worm-based propagation for the majority of infections. This suggests that IoT is not yet well understood by users and is hence lacking the required security measures. Whilst the choice for simple propagation methods may be related to device limitations, this also makes IoT an attractive target for botmasters despite device constraints. If nodes' power and other resources are well managed, the botnet can quickly infect many low-capacity devices, resulting in a sizable attack force. This was demonstrated in the case of Mirai [1] and will likely be seen in its derivatives as well.

3. Proposed Model

We developed `IoT-SIS`, a novel propagation model to explore the characteristics of IoT networks and the botnets targeting them. The model's starting scenario assumes

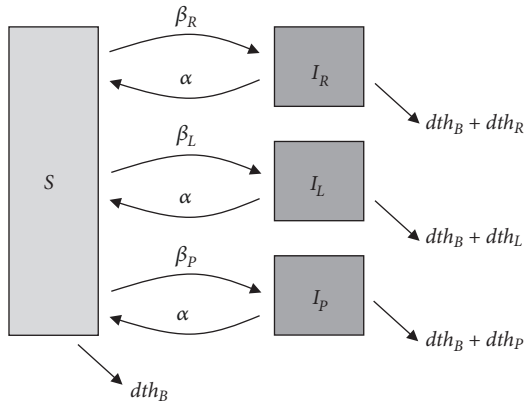


Figure 1: Flow diagram of proposed IoT-SIS model with states and key transitions.

that there is an existing infection and seeks to measure how quickly and widely this might spread. Note that we specifically focus on battery-powered IoT devices in order to understand the impact of energy consumption in this context. For simplicity, we make the following assumptions: (1) a static network, i.e., nodes have no mobility, (2) a deterministic deployment layout, i.e., static and predefined, (3) node homogeneity, i.e., nodes consist of similar devices with the same functionality and properties [9], and (4) a mesh topology, i.e., nodes are connected directly to many other nodes, unlike in a star topology. We do not consider the source of infection, or the impact of sleep cycles (the schedules of which will be different for each scenario).

IoT-SIS is based on the *SIS* paradigm from epidemiology. Given a population of nodes N, a set of compartments represent possible node states. Then, differentials are added to describe the rates of change between the proportional node populations. Susceptible S nodes are free of malware but vulnerable to infection; this is the default state of nodes. Infected I nodes are carriers who can transmit the infection. Given a series of time-steps represented by t, the number of nodes within each state is a fraction of N such that

$$N = S(t) + I(t) = \frac{S}{N} + \frac{I}{N} \tag{1}$$

Successful contacts, followed by successful transmission events, cause nodes to pass from S to I. The I nodes are categorised based on the nature of the infection event which led to their transition. The model is illustrated in Figure 1 and mathematically defined by the following system of differentials:

$$\frac{dS}{dt} = -\beta_R SI - \beta_L S_{loc} I - \beta_P S_{nhb} I - dth_B S + \alpha I \tag{2}$$

$$\frac{dI_R}{dt} = \beta_R SI - \alpha I_R - dth_B I_R - dth_R I_R \tag{3}$$

$$\frac{dI_L}{dt} = \beta_L S_{loc} I - \alpha I_L - dth_B I_L - dth_L I_L \tag{4}$$

$$\frac{dI_P}{dt} = \beta_P S_{nhb} I - \alpha I_P - dth_P I_P - dth_P I_P \tag{5}$$

where

(i) S is the total susceptible population.

(ii) S_{loc} is the fraction of S in the local network (of a node).

(iii) S_{nhb} is the fraction of S in the neighbour set (of a node).

(iv) I is the total infected population, $I = I_R + I_L + I_P$.

(v) I_R is the fraction of I nodes infected via random scanning.

(vi) I_L is the fraction of I nodes infected via local scanning.

(vii) I_P is the fraction of I nodes infected via P2P.

(viii) β_R is the random scanning-based infection rate.

(ix) β_L is the local scanning-based infection rate.

(x) β_P is the P2P-based infection rate.

(xi) dth_B is the death rate due to standard activities.

(xii) dth_R is the death rate driven by random scanning.

(xiii) dth_L is the death rate driven by local scanning.

(xiv) dth_P is the death rate driven by P2P communication.

(xv) α is the recovery rate.

Nodes are lost from the S state at the rate of infection. Infections may be based on random scanning (R), local scanning (L), or P2P communication (P). Nodes transition into the I subset which aligns with their infection type. Nodes are lost from the I state at the rate of recovery and transition back into S. We also consider death rates. Nodes may die of 'natural causes' at the benign death rate dth_B. Nodes may die due to malicious activity at the malicious death rates. These parameters are described in more detail in the following subsections.

3.1. Network Structure & Population. Conceptually, the network is made up of multiple M interconnected WSNs. The total population N is divided amongst these WSNs:

$$N = WSN_1 + WSN_2 \ldots WSN_M \tag{6}$$

These WSNs may be connected directly, via IP-based infrastructure networks, or via the Internet. For simplicity, we do not define a separation between these connectivity types and assume that infection type is more relevant. Hence, the model view encompasses the collection of WSNs. We do consider the type of infection associated with the different connectivity types as follows:

(i) **Inter-WSN**: traffic is routed over layer 3, with targets found via random scanning.

(ii) **Intra-WSN:** traffic is routed over layer 3, with targets found via local scanning.

(iii) **Between neighbours:** traffic is exchanged via P2P over layer 2, with targets based on P2P relationships.

This means that each infection method has a different attack surface, consisting of different proportions of the S population. If a given I node uses random scanning, it has

access to all of S, whereas with local scanning it can only reach the proportion of S which is local to it. Similarly, this node can only reach its direct neighbours if it uses P2P-based propagation. This is detailed further in the coming sections.

We assume that the population has a finite number of nodes; i.e., there are no births into the system. Nodes may be removed from the population via deaths, either 'naturally' due to standard end-of-life or wear-and-tear or as a direct result of bot infections. Natural deaths occur in both S and I populations, whilst bot deaths only occur within the I population.

3.2. Infection Rates. There are 3 infection rates, each representing a different propagation method. For each, the infection rate β is

$$\beta = contact\ rate \times P_{transmission} \quad (7)$$

where $P_{transmission}$ is the transmission probability (i.e., the infection probability per contact). This is sampled from a Poisson distribution, where λ is the mean number of successful transmissions of the infection per contact per time for a single I node. Hence, it is the proportion of total contacts per time which lead to infection. The Poisson distribution is applicable where events are discretely measured, and where event occurrence is rare per given period [13]. Transmissions can be counted discretely as individual events and, given a wider network of WSNs, should be relatively rare. Furthermore, the probability of an I node causing an infection does not change over time, and previous successful or unsuccessful attempts do not impact the chances of future attempts [13].

Additionally, any increase in the number of infections is caused by the growth of the I population rather than an increase in the infection rate. Hence, transmission events are independent. Given the scanning rates of known botnets, we can estimate the proportion of contacts which result in successful transmissions and, hence, estimate a value for λ. The total I population is split into 3 subsets to match the infection methods: random scanning, local scanning, and P2P. Note that these sets represent how the nodes became infected, but any I node may perform any kind of infection.

Theoretically, a worm-based bot malware may use 1, 2, or all 3 of these infection types. Random scanning of the IP address space has been observed frequently in worm-based propagation, e.g., Mirai [2]. Meanwhile, HideNSeek reportedly changes its behaviour if the infected IP is within the same LAN as the infecting source node [4]. When local, a TFTP connection is used to download the malware from the source node. Otherwise, it must be downloaded remotely [4]. Additionally, HideNSeek is described as a P2P botnet [4]. IEEE 802.15.4 allows P2P communication between neighbour nodes. Meanwhile, users often are not aware of the full functionality of their IoT devices. This means that it is feasible that IoT-targeting bots will further exploit P2P as a contact vector.

3.3. Propagation Mechanisms. Scanning-based propagation requires nodes to make connections to remote nodes which they then attempt to gain access to. However, connection attempts may be unsuccessful because the IP does not correspond to an active node, the target device is not running targeted services, or due to simple network error. Hence, the random-scanning contact rate is defined:

$$contacts\ per\ time = scans\ per\ time \times P_{success} \quad (8)$$

where the $P_{success}$ is the probability of connection success, sampled from a Poisson distribution with λ defined as the mean number of connection events. Random-scanning behaviour will target the whole S population, including remote WSNs, and hence it is feasible to assume that this will result in some unsuccessful connections. In contrast, we would expect more local scans to be successful and most P2P connections to be successful. Therefore, this is reflected in the definitions of the infection rates.

3.4. Subsets of S Population. Given a network space made up of multiple WSNs, random scanning relies on routing to scan the entire IP address space and hence targets the whole S population (i.e., all S nodes across all the WSNs). Local scanning is similar but targets the local IP address space (i.e., within a single WSN). Directly connected neighbours use P2P, and hence P2P-based propagation targets only a node's neighbour set. In short, each infection method has access to a different proportion of the available S population. Where *loc* is the mean number of nodes in 1 average WSN, we define S_{loc} as the fraction of the total S population within a local network. This determines the attack surface of local scanning. Similarly, where *nhb* is the mean number of nodes within the neighbour set of 1 average node, S_{nhb} is the fraction of the total S population which makes up the attack surface of P2P infections.

We find the fraction of N within each WSN (assuming that all WSNs are of the same size) and then take that percent of the current S population to find the final S_{loc} value. This means that the target population accessible via local scanning is always capped. Similarly, we find the fraction of N within each neighbour set and then take that percent of the current S population to find S_{nhb}. The mean size of a P2P neighbour set will be a function of the node distribution scenario, the average node transmission range, and the total nodes per WSN.

3.5. Deployment Setups & Neighbour Sets. Before RPL arranges nodes into DoDAGs for routing and IP connectivity, each node can form layer 2 P2P relationships with neighbours who are within its transmission range. Hence, each node has a P2P neighbour set associated with it. We make the following assumptions:

(i) All nodes have the same transmission range.

(ii) Nodes are uniformly distributed (i.e., equally spaced).

(iii) WSNs have uniform node counts and deployment areas.

For a uniform distribution, the node density (per unit of space) is defined by (9). Given the node transmission range, the number of neighbours per node is given by (10).

$$density = \frac{total\ no.\ of\ nodes}{deployment\ area} \quad (9)$$

$$no.\ of\ neighbours = density \times transmission\ range \quad (10)$$

3.6. Deaths. We assume that the energy spent on sensing and processing is negligible in comparison to the energy spent on sending and receiving transmissions. Hence, we focus only on the energy consumed for communication in this model. Significantly, this will be directly impacted by the propagation activities of bot-infected nodes. Excluding random errors or physical tampering, nodes deaths are caused primarily by power depletion. All nodes consume energy when sending/receiving traffic which is part of their normal operation. Hence, the benign (i.e., 'normal') death rate depends on the normal contact rate of nodes.

If we assume that bot nodes send and receive additional traffic, then the contact rate of infected nodes should be higher than for S nodes. Therefore, alongside the normal contact rate, we also introduce a malicious contact rate. Death rate caused by malicious behaviour then depends on the malicious contact rate. Propagation can be attributed to this additional traffic, and subsequently the infection rate depends on the malicious contact rate as well. Hence, bot nodes should die at standard rate plus the malicious rate. Node death rate (*dths*) is dependent on the amount of transmitted data, the transmission distance, and the characteristics of the node as follows:

$$power_{msg} = \mu \times mean\ message\ size \times distance \quad (11)$$

$$power_{time} = power_{msg} \times contact\ rate \quad (12)$$

$$node\ lifespan = total\ battery\ capacity \times power_{time} \quad (13)$$

$$dths = \frac{1}{node\ lifespan} \quad (14)$$

where the μ is the mean power needed to transmit 1B of data 1m in distance, $power_{msg}$ is the power required per message, $power_{time}$ is the power consumed per time, and *distance* refers to how far apart nodes are spaced, defined by

$$distance = \frac{deployment\ area}{no.\ of\ nodes} \quad (15)$$

Given the mesh setup of the sensor network, nodes should always forward traffic to an immediate neighbour, regardless of whether the destination node is nearby or in a remote WSN. Hence, exchanging data with an immediate neighbour will consume as much power as exchanging data with a remote node. As benign and malicious activities have separate death rates, there may also be different contact rates and message sizes associated with them. Since the infected death rate is the inverse of the infected lifespan, a longer lifespan should result in a lower death rate and vice versa. If an I node performs no malicious activity (malicious contact rate=0), it lives out the normal lifespan. If an I node performs no normal activity (benign contact rate=0), it lives out the infected lifespan. If an I node performs both types of activity, its lifespan is shortened, proportional to the contact rate of each traffic type. Therefore, bot nodes are more likely to die earlier. Note that each infection type has its own death rate, driven by the relevant contact rate.

3.7. Recovery. The recovery rate α depends on the behaviours of network defenders, including how active they are in their monitoring and how effective they are at identifying and cleaning I nodes. Hence, it is plausible for us to control the recovery rate directly to determine what level and type of engagement is necessary to effectively mitigate botnet spread. For instance, we can increase or decrease recovery rate to observe the impact of faster or slower interventions. Therefore, α can be estimated by the user of the model. Since all the nodes in a single WSN will be serving the same purpose, it is plausible to assume that they are of the same hardware and running the same software [14]. Hence, a single patch type should address infections caused by a single worm strain.

4. Simulations

4.1. Setup. Our aim was to test the population dynamics and parameter relationships in the model under different conditions. To achieve this, we ran a series of manual numerical simulations with different input values from which we gained an understanding of these relationships. This was followed by a formal Monte Carlo simulation to generate a range of outputs and to understand the likelihoods of these outcomes based on our starting assumptions.

Botnets have particular requirements. For instance, they usually need to collect as many bots as possible to have a sufficient attack force. They then need to be able to sustain that population over time in order to launch successful campaigns. To address this, the model principally covers 2 planes: space and power. The spatial aspects of IoT-based bot propagation relate to the infection types and their corresponding attack surfaces. This determines the reach of the infection. The power aspects are addressed through the contact rates and the death rates. This determines the power depletion applied on the nodes. This setup also aligns with 2 defining characteristics of IoT networks, which are dense node deployment and limited power availability. Based on the model's 2 planes, we predicted the following:

(i) There is a significant relationship between infection rate and malicious death rate.

(ii) Power dynamics will be different for dense infected networks vs. sparse infected networks.

(iii) There is a relationship between propagation attack surface, the bot count, and the spatial distribution of bots.

The model itself is programmed as an R script [15]. We chose to use R because it is both powerful and free,

Table 2: Inputs for the Monte Carlo simulation.

Parameter	Input	Explanation
Message size	Normal dist. sample, mean=50B, sd=5B	Mirai scans less than 250B/sec [2]. Scales with contact rate.
Max power	864000mAs	Typical sensor battery capacity of 240mAh [12]. Converted into mAs.
Power used	{0.5mA, 0.75mA, 1mA}	To send 1B 1m. Estimated from 30mA peak for node [12].
Contact rates	{1/s, 10/s, 20/s}	IoT-based worms should be relatively slower than standard worms.
Recovery rate	{0.25, 0.5, 0.75}	Estimated by us based on degree of security engagement.
WSN count	{1, 5, 10}	Range of sensor networks available.
Trans. range	{10m, 100m}	Typical range is 10m, we also test larger theoretic range of 100m.
Deployment area	$\{50m^2, 100m^2\}$	Small and large deployment regions. Will change with application.

allowing for complex operations and making our process easily repeatable by others. The script uses the `deSolve` package [16] to solve the differential equations in the model system and the `MonteCarlo` package [17] to run the simulations. Table 2 summarises our input ranges. Due to the costly nature of running Monte Carlo simulations, and the number of parameters included in the model, the input range for the Monte Carlo process must be limited, whilst keeping the values meaningful. Our approach here is inspired by [18]. The aim is to include high and low values to cover a wide-enough range of possible outcomes. The Monte Carlo outputs were averaged over 100 iterations, and each run of the model goes through 100 time-steps. The results of a relationship analysis and simulations are detailed in the next subsection.

4.2. Results. Figures 2(a), 2(b), and 2(c) show the Monte Carlo histograms for the sizes of the infected populations at the final time step. As expected, random scanning has the largest impact on the network due to having the largest attack surface. Hence, the I_R population is capable of consuming the whole network. The simulation also showed that I_R becomes negative if no infections are taking place, due to the consistent death rate. Meanwhile, the I_L and I_P populations have a more modest impact due to their capped attack surfaces.

Using only random scanning (resulting in $I = I_R$) allows the infection to spread throughout the S population, with higher peak infections thanks to there being no other I subsets. This mimics the dynamics of typical worm-based propagation. Meanwhile, local scanning on its own (resulting in $I = I_L$) struggles to have a significant impact if its attack surface is too limited. Hence, I_L requires (a) an increased contact rate, (b) a large transmission probability, or (c) fewer WSNs (i.e., individual WSNs must be large enough).

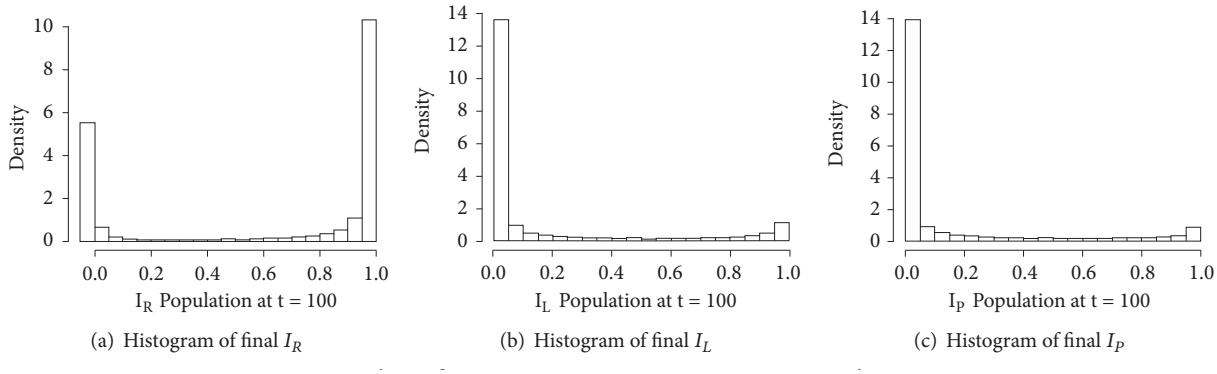
Similarly, using only P2P (resulting in $I = I_P$) leads to a more extreme version of the I_L scenario, as I_P is capped even more. Hence, I_p also requires (a) increased contact rate, (b) increased transmission probability, and (c) fewer WSNs, but it can also be boosted by reducing the size of the deployment area or increasing the node transmission range (i.e., neighbour sets must be large enough).

Larger message sizes require more power to transmit and hence drive up the death rate. Meanwhile, increasing nodes' battery capacity can increase nodes' lifespans and hence decrease the number of deaths per time. A larger deployment area decreases the overall density of nodes, increasing the distance data must be sent over and consequently causing more deaths. Figure 2(d) depicts this, where dth_B decreases as density increases. Meanwhile, higher density results in a higher probability of a large final I population, as illustrated in Figure 2(e).

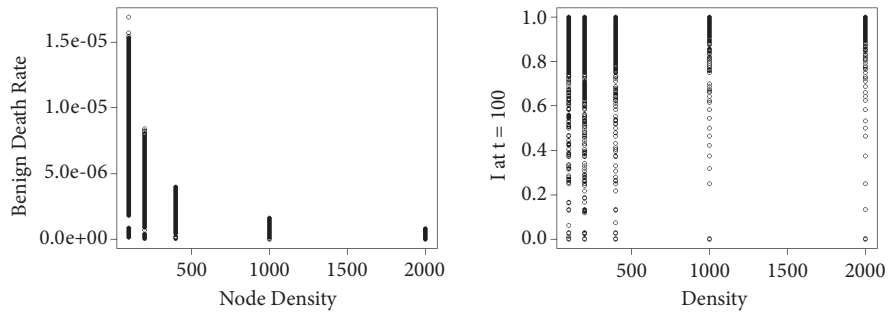
A larger deployment area also decreases S_{nhb}, leading to a smaller I_P population as the available peer nodes are minimised. Meanwhile, S_{nhb} grows with the node transmission range and allows I_P to reach higher peak values thanks to the expanded attack surface. A greater number of WSNs shrinks S_{loc} and S_{nhb}, as well as the density of each WSN. This pushes up the distance between neighbouring nodes, which in turn consumes more power to transmit data. This results in higher death rates, and a smaller final N, as Figure 2(f) illustrates. Conversely, dense node deployment results in lower death rates as data is transmitted over shorter distances. The sizes of the infected populations are also impacted.

Increasing or decreasing the contact rate has a corresponding impact on the associated death rate. A larger benign contact rate pushes up dth_B leading to more deaths overall. For bot contact rates, changes also influence the associated infection rate. Given N, the I population is distributed amongst the 3 infected subpopulations depending on β and the number of available S nodes. Hence, when 1 subpopulation increases in size, the others shrink proportionally. The overall impact of increasing contact rates is a sharper increase in infections, followed by a sharper decline in the I population due to a larger number of deaths; i.e., the I population is not sustained. This is demonstrated in Figure 3(a), where a higher contact rate pushes up the death rate until the population becomes depleted.

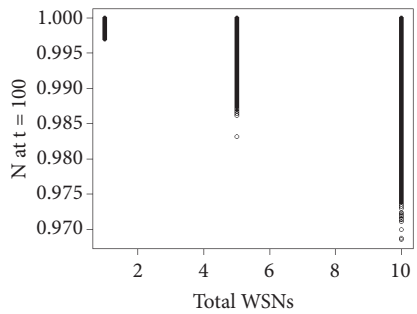
The capped propagation methods are more sensitive to drops in contact rate since they are already handicapped. I_L and I_P populations may overcome this handicap and surpass I_R if their corresponding infection rates become very large. Figure 3(b) demonstrates this effect for I_P, with increasing contact rates across 5 simulations (in ascending order so that *sim*#1 used a contact rate of 5/s and *sim*#5 of 1,000/s). Meanwhile, changes to $P_{transmission}$ only impact infection rates and not death rates because it changes the proportion of contacts which result in infections (whilst the contact rate remains constant). Hence, the I subpopulations reach higher



(a) Histogram of final I_R (b) Histogram of final I_L (c) Histogram of final I_P



(d) Node density vs. benign death rate (e) Node density vs. final I



(f) Total WSNs vs. final N

Figure 2: Histograms and parameter relationships from the Monte Carlo simulation.

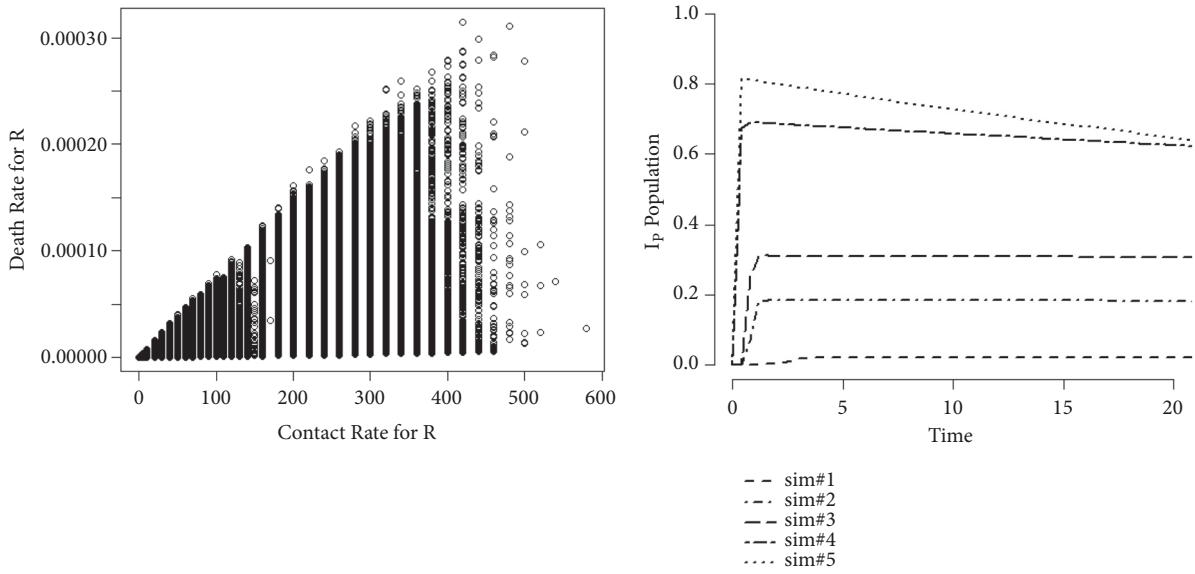
peak values faster, whilst the downward slope caused by deaths remains constant.

5. Discussion

5.1. Implications for Malicious Actors. Our findings have a number of real-world consequences. In this section, we analyse the simulation results to identify how botmasters may achieve better propagation results.

We considered propagation methods with access to different proportions of the S population in order to get a macroscopic view of a larger sensor node population. (Conversely, if the model scope was at the local or neighbour level, we would have a microscopic view of those populations.) In doing so, we found that local and neighbour set infections tend to remain endemic and do not have significant impact on the larger population, unless the attack force or node density is very large. In parallel, these factors also impact power usage and hence the network lifetime. All of this means that a lower contact rate with further reach is more potent than a higher contact rate with a shorter reach. This is significant for botmasters, who will prefer a propagation method which achieves the largest increase (with the widest spread) in the shortest time.

This also highlights the role of Internet connectivity in wide-scale propagation. Conventional WSNs are not necessarily connected to the Internet, but IoT-based WSNs are. This means that they are exposed to infection via random scanning, so that existing bots can target vulnerable nodes in remote WSNs. This means that a botmaster can overcome (a) the limited attack capacity of individuals and (b) the limited propagation activity of individuals, by accumulating more bots. Large-scale deployment of IoT-based WSNs is likely to increase in the future, for example, with the advent of smart cities. With this kind of wider adoption in industry and infrastructure, there is a prominent risk of these large-scale networks being targeted by botnet malware.



(a) Contact rate for R vs. death rate for R (dth_R)

(b) I_P where P's contact rate = {5, 50, 100, 500, 1000}

FIGURE 3: Plots showing the impact of contact rates in numerical and Monte Carlo simulations.

More aggressive propagation campaigns will consume more energy, such that an overaggressive strategy may become inefficient as nodes die at a rate equal to or greater than the rate of infection. A slow decline in the I population means that the botnet population is steady. Hence, botmasters are likely to gain a larger botnet via random scanning when they can keep contact rate at a reasonable level. Botmasters may try to maximise the contacts-to-infections ratio (via the transmission probability) to maintain a larger number of bot nodes without driving up death rates. Furthermore, propagation messages should remain small to consume less energy during transmission. This will also be relevant for continuous C&C traffic.

Node density plays a central role in network lifetime. Denser networks were shown to consume less energy at the individual node level to send and receive data. This, combined with the larger number of possible bots, makes dense WSNs more desirable targets for botnet formation. WSNs with a lot of activity may be attractive if node hardware is of a higher specification to deal with this (e.g., higher battery capacity or larger transmission range). If the WSN only contains average nodes, however, a highly active network will probably be undesirable for botmasters as network lifetime will be compromised. A possible solution to this would be the addition of bot functionality which cancels scheduled tasks to reduce power depletion.

5.2. Implications for Defensive Actors. Now that we have discussed our results from the botmaster's perspective, we go on to consider how our findings can provide insight for defenders and improve the security of IoT networks.

When implementing WSN-based security provisions, individual nodes do not usually possess the processing power or energy capacity for host-based detection. Therefore, a network-based mechanism is required. The simulation results showed that propagation dynamics can change at different community levels (i.e., inter-WSN vs. intra-WSN vs. between neighbours). Therefore, it may be beneficial to add network monitors at each level to identify small-scale spread or endemics sooner. This could form the basis for an IoT-specific variation of the defence-in-depth security paradigm.

In reflection of the `IoT-SIS` model's outputs, detection approaches should prioritise instances of random-scanning behaviour, as this was shown to increase the chance of an epidemic. This type of scanning can be characterised by a pattern of probing behaviour (to identify the presence of a worm) combined with a large number of outgoing connections and the indiscriminate selection of destination IPs. Furthermore, detection methods would benefit from considering slower scanning as we identified that keeping contact rate minimal can result in better bot node retention. Slow scanning rates can also be used as an obfuscation method by botmasters.

Alongside the data they collect from the environment, sensors also generate and share telemetry data which describes a node's status (e.g., current engagement, current location, role in the current topology, and power levels) as well as various details on its communications with other nodes. The `IoT-SIS` model demonstrated that nodes are likely to deplete their finite power resources more quickly when they become bots. Hence, telemetry data can be used to monitor nodes' battery levels, delays in the execution of scheduled tasks (aimed to conserve power), or patterns of anomalous communication behaviour caused by botnets' reliance on automation.

Furthermore, as end-to-end encryption becomes more widely adopted (by both defensive and malicious actors), detection systems will need to focus on telemetry data as payload examination becomes unfeasible under these conditions. Such data for propagation detection may include

message lengths, connection durations, various timestamps, power levels, and, where relevant, GPS coordinates. Since bot traffic is repetitive and systematic by nature, detection can be achieved in encrypted networks through pattern identification in this telemetry data.

When developing immunisation schemes to deal with ongoing epidemics, a common approach is to minimise the frequency of contacts between infected and susceptible nodes to reduce the number of new infections. However, we found that propagation can be more successful if the contacts-to-infections ratio is maximised instead of the contact rate which may (a) cause more node deaths and (b) reveal bots' presence to defenders. To reduce the number of successful infections, defenders should focus on securing individual nodes via proper login credentials (changed periodically), updated and fully patched software, and the disabling of unnecessary services. This is aimed at reducing the transmission probability, and experience shows that such simple steps could mitigate existing IoT worms like Mirai [2]. We believe that this approach is more pragmatic than blocking connections or taking nodes offline which can have a negative impact on routing and network convergence. Additionally, immunisation and recovery efforts should be applied across all potentially targeted WSNs for effective mitigation, as random scanning was shown to be able to drive the bot malware successfully across multiple WSNs for wide-scale coverage. Hence, immunisation must be equally widespread in its scope.

We should consider dense networks of high-grade sensors to be particularly desirable for botmasters. P2P communications should be well controlled and monitored to ensure that compromises to the local network have limited impact. In extreme cases (and where functionality is not affected), P2P may even be disabled entirely. This may be particularly relevant for home environments consisting of few or individual sensors (rather than a set of collaborative sensors).

Our results suggest that multiple small but dense WSNs (with minimal P2P contact) are better at preventing bot epidemics. In our model scenario, the denser the network, the lesser the energy used by individual nodes (causing fewer node deaths) and the smaller the attack surface (capping the reach of the malware). However, in real-life, this should be considered in the context of the features of the given network, including its application and the protocols used, in order to avoid generalisations which overlook the particularities of different scenarios. Furthermore, the suggestion to use small, dense WSNs needs to be balanced with the routing performance and application requirements specific to each scenario.

Since propagation is a difficult process for detection in real-time, as part of our future work, we would like to explore how propagation models such as ours can be aligned with real-life networks and measurements of traffic to create an application framework. The framework would aim to help users yield meaningful predictions and to aid early detection. Furthermore, we have taken a simplified macroscopic view of IoT networks in this model. However, research suggests that sink nodes are more vulnerable to power depletion due to their role as a gateway for all incoming/outgoing traffic. Hence, in our future work, we would like to consider the role this plays in botnet propagation.

6. Related Work

6.1. Existing Propagation Models. Malware propagation is a difficult but significant process to observe and measure to be able to effectively tackle the threat of cybercrime. Hence, there is a range of literature on the subject, and despite most of it dealing with malware in conventional and mobile networks, there has been a push in recent years to expand this into the analysis of WSNs and IoT. Proposed models tend to follow the state-based transition approach provided by epidemiology given its clear definitions and simple structure—a trend that this work also follows. WSN-based models focus on particularities of the environment, like node mobility, transmission range/radius, topological variances impacting node density, and energy usage, alongside more typical factors like user awareness and recovery rates. The following is a selection of existing research chosen to demonstrate the state-of-the-art and to provide context for our work.

Wang et al. [14] designed a state-based model to observe worm propagation in WSNs with mobile actuators. The authors suggest that actuators can increase the speed of worm spread if successfully compromised. The model probabilistically estimates node states to "microscopically compute the prior probability" of individual sensor infections via directly connected neighbours [14]. Nodes may be susceptible (S), contagious (C), or infected (I) and are deployed with an infected mobile actuator moving randomly amongst them. The infection's spatial distribution is defined based on I node locality. This includes the identification of S nodes with and without infected neighbours. The energy consumed by the network is calculated based on a percentage increase in consumption in individual nodes after infection. The model was simulated and compared to others and reportedly produced different results for different node density values. Overall, the authors report that the inclusion of an infected mobile actuator improved worm propagation across the test scenarios, including high and low-density cases. The mobile worm was also found to increase energy consumption when compared to similar static worms.

Ji et al. [19] focused their work on Mirai's architecture to study its propagation patterns. They based their propagation model on the *SIR* format, with N defined as the total IoT population, S as IoT nodes with weak logins, I as infected nodes, and R as immune nodes. The online device count is S multiplied by the rate that devices come online α, whilst the infection count is the product of α and I. The attack surface is the product of α and $(N - P)$, where P is the portion of the address space to be ignored. The authors state that because Mirai bots did not infect targets directly (using the loader instead), the model should assume that IoT devices do not cause secondary infections. They also suggest that increase in the I population will increase traffic load for the network. Hence, they propose that the infection rate will decrease as I grows. Using simulations based on estimates of the US IoT-enabled camera population, they report that I increased

steadily for a time before slowing down, due to the depletion of 'easy' targets.

Singh et al. [20] studied worm propagation in WSNs using the *SEIRV* model, where E denotes nodes which are infected but not infectious, who transition into the infectious state I, and V denotes vaccinated nodes who are permanently immune. Nodes are uniformly distributed with a transmission range of r, and the transmission region of individuals is calculated based on this. The authors use the basic reproductive number R_0 to define equilibrium points and then to evaluate the system's stability at these points. Through this approach, they define thresholds for both the transmission radius and the density and then use these to test different values for these 2 parameters. They were then able to demonstrate the relationships between the thresholds and the equilibrium points; i.e., epidemics fail when the radius or density is less than the threshold and are successful otherwise.

Gardner et al. [21] developed the IoT-BAI (IoT Botnet with Attack Information) model, based on the *SEIRS* format. The model is grounded on the propagation dynamics of Mirai-like malware and considers the impact of user behaviours, specifically in relation to increased awareness following a publicised attack. Nodes may transition into the R state from any other state, and the rates of recovery increase for a finite period following an attack. Hence, there are 2 sets of recovery rates; one during propagation ("Botnet Growth Phase") and one during the increased awareness period ("Botnet Reduction Phase") [21]. The IoT-BAI model does not consider deaths but does incorporate a constant birth rate to reflect the growth of the IoT device population [21]. Based on their simulations, the authors suggest that the constant stream of new devices makes the IoT network increasingly vulnerable. Meanwhile, greater user awareness triggers the Botnet Reduction Phase sooner and increases the time between epidemics, theoretically reducing botnet impact.

Mishra et al. [22] created the *SEIRS-V* model (where *S-V* represents susceptible nodes who have received a vaccination) to study the propagation of worms in WSNs. The model includes births (A) as well as deaths. Two separate death rates are defined: μ for standard hardware/software failures and ε for device failures caused by worm infections. Birth and death rates are such when there is no malware, the population size can be estimated as the 'carrying capacity', which is defined by A/μ. There are also separate rates for recovery and vaccination and, consequently, there are separate immunity periods associated with each. The authors use MATLAB for simulations and show that greater emphasis on recovery and vaccination can significantly mitigate the scale of infections by absorbing more S nodes into the R and *S-V* compartments. Hence, susceptibility of nodes to future infections is reduced. They also stress the expansive applications of WSNs in various areas of industry and healthcare, highlighting the need for effective malware defences.

Feng et al. [23] used the *SIRS* model to consider worm propagation in WSNs with a focus on nodes' transmission radius, energy consumption, and the network density. The model assumes a uniform distribution of nodes in a 2D space, and nodes may recover from both S and I states. Nodes may die in each compartment due to power depletion at the defined death rate, whilst some recovered nodes may probabilistically become susceptible again. Based on the reproductive ratio R_0, the authors define a threshold for the transmission radius such that for a value lower than this threshold and with $R_0 \leq 1$, the worm cannot survive. Similarly, they define a threshold for node density such that for a value lower than this and with $R_0 \leq 1$, a "worm-free equilibrium" is maintained [23]. Through numerical simulations they consequently demonstrate that a smaller transmission radius or lower network density can mitigate worm propagation.

Jerkins et al. [24] used the principles of epidemic modelling to boost the security of IoT devices via "inoculation epidemics" using the *SI*/*NS* (Susceptible, Infected/Non-vulnerable, Susceptible) model. They aimed to use a process similar to malware propagation (via the *SIS* model) to identify vulnerable nodes and patch them. A 'vaccine' is developed by reverse-engineering captured malware, specifically focusing on the infection vectors and exploits used. The vaccine then propagates like a worm using the same methods to deliver a patch, thus giving nodes immunity against that malware. In the model, N denotes nodes which are 'infected' by the vaccine. Separate infection rates are defined for the malware and the vaccine, such that an epidemic fails when the vaccination rate is greater than the infection rate, and vice versa. Additionally, nodes may reboot at a rate of β for I nodes and θ for N nodes, such that an epidemic fails if $\beta > \theta$. Through simulations, they demonstrated that increasing the number of nodes which are vaccinated against the malware mitigates its propagation and diminishes its overall impact.

6.2. Comparison to the Proposed Model. Each of these works approaches the study of WSN-based worm propagation in a different way. Wang et al. [14] focus specifically on the mobile actuator scenario, demonstrating how the IoT space may present unique vulnerabilities and exploitation opportunities. Ji et al. [19] provide a specialised model for the Mirai botnet, driven by a need to understand the propagation of this prevalent threat. Singh et al. [20] and Feng et al. [23] focus on defining R_0-based thresholds, using the epidemic-based metric to determine the limits of spread. Both Gardner et al. [21] and Mishra et al. [22] emphasise recovery, considering the impact of user behaviours and vaccinations, respectively. Meanwhile, Jerkins et al. [24] presented a novel approach by appropriating epidemic processes for defence.

Most of these works study the spread of WSN-based worm malware, with far fewer focusing on the presence of botnets within WSNs. Botnets are different to worms (despite sometimes being spread in a worm-like manner) primarily because the retention of infected nodes is crucial to the goals of the botmaster. A sufficient number of bots must be accumulated for the botnet to be effective, whereas a worm has no such requirement. The proposed model aims to capture this and to explore the surrounding factors. The works of Ji et al. [19] and Gardner et al. [21] are botnet-focused, but they concentrate exclusively on Mirai, using empirical measurements of this malware to build models which characterise it specifically. In contrast, the

proposed model considers a wider range of behaviours by incorporating observations from Mirai's descendants as well. We believe that this expands the models applicability as a result. Furthermore, the scenarios used by the related works are based on standard wireless sensor nodes, whereas our model is designed specifically with IoT-based sensors in mind the difference being that IoT devices have constant Internet connectivity.

These papers also have vague or abstract definitions of scanning behaviour, assuming a standard approach across all bot instances. An exception to this is Ji et al. [19], who consider the attack surface to be defined based on the infection rate and omitted portions of the IP address space. Meanwhile, the `IoT-SIS` model defines 3 separate and clearly defined scanning behaviours, along with the corresponding attack surfaces that become available with each method. This is justified based on observed bot behaviours. Mirai is known to engage in random, global scanning (omitting certain known IP ranges) [1, 2], and HideNSeek was observed to scan locally, changing its methods when source and target nodes were in the same LAN [4]. Finally, sensor nodes have P2P connectivity, which is a well-established botnet propagation method in conventional networks, and so should be considered in the IoT context as well.

We found that death rates were used inconsistently across existing propagation models, with varying levels of importance placed on this transition. Ji et al. [19] and Gardner et al. [21] did not consider death rates, despite their empirical focus on Mirai. Mishra et al. [22] and Feng et al. [23] did incorporate death rates, with the former being somewhat similar to our work because separate death rates are included for both normal and malicious processes. However, neither of these works is in the context of botnets, and hence they do not consider the effect of deaths on botnet formation. We address this and additionally provide a definition of the death rate based on node characteristics and communication behaviours. Furthermore, the proposed method aims to explore the relationship between deaths and propagation activity by relating energy depletion to scanning behaviours via contact rates. The rationale is that contact frequency determines the amount of node energy consumed for communication, whilst also determining the number of possible infections based on contacts between S and I nodes. We believe that this has not been demonstrated before.

By making the model specific to IoT-based botnets and incorporating different types of scanning and deaths, we were able to identify some dynamics which, to our knowledge, have not been presented in the existing literature. Simulations of the `IoT-SIS` model showed that the propagation method and the available attack space impact the spatial distribution of bot nodes, such that methods limiting spread to nearby nodes tend to cause intra-WSN endemics rather that epidemics. We also found that driving the malware to propagate faster/harder causes nodes to consume more of their finite energy, thereby endangering the longevity and consistency of the botnet. Based on this, we were able to determine that propagation strategies in IoT networks are more effective if the transmission probability can be maximised instead of the contact rate.

Therefore, this paper sits alongside existing works (such as those discussed here) by providing a general model of worm-based botnet propagation in WSNs to explore the key characteristics of IoT networks at a macroscopic level. Our model is not based on a specific scenario, a specific malware strain, or aimed at deriving particular measurements. Instead, we explore the factors at play in botnet formation. We believe that there is a need for this kind of approach due to the unique features of botnets that set them apart from other types of malware and the unique features of IoT networks which give rise to different environments and scenarios to those of conventional networks.

7. Conclusions

In this paper, we have developed the novel `IoT-SIS` botnet propagation model focused on IoT sensor networks and explored how the IoT-specific characteristics may impact botnet formation. We were able to improve our understanding of the botnet threat amongst sensor devices and to explore the relationships between network density, node power, scanning behaviours, and attack surface size for different scanning methods. Our simulations showed that dense networks allow better distribution of activity, resulting in longer lifespans for individual bots, and that aggressive propagation approaches can be counterproductive in procuring nodes. We also showed that scanning rates and transmission probability must be increased significantly in order to overcome capped S populations. In the continuation of this research, we hope to explore ways to improve the accuracy of propagation models and to better align compartmental models with network-based traffic analysis. In future models, we aim to look more closely at particular IoT-specific phenomena, including the rapid power depletion of sink nodes, the use of encryption, and the impact of mobility.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," *IEEE Computer Society*, vol. 50, no. 7, pp. 80–84, 2017.

[2] M. Antonakakis, T. April, M. Bailey et al., "Understanding the Mirai Botnet," in *Proceedings of the USENIX Security Symposium*, pp. 1092–1110, 2017.

[3] G. Kambourakis, C. Kolias, and A. Stavrou, "The Mirai botnet and the IoT Zombie Armies," in *Proceedings of the IEEE Military Communications Conference, MILCOM 2017*, pp. 267–272, IEEE, October 2017.

[4] B. Botezatu, "New HideSeek IoT Botnet using Custom-Built Peer-to-Peer Communication Spotted in the Wild," https://labs.bitdefender.com/2018/01/new-hide-n-seek-iot-botnet-using-

custom-built-peer-to-peer-communication-spotted-in-the-wild/.

[5] W. O. Kermack and A. G. McKendrick, "A contribution to the mathematical theory of epidemics," in *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, vol. 115, no. 772, pp. 700–721, 1927.

[6] L. J. Allen, F. Brauer, P. Van den Driessche, and J. Wu, *Mathematical Epidemiology*, vol. 1945, Springer, 2008.

[7] J. O. Kephart and S. R. White, "Directed-graph epidemiological models of computer viruses," in *Computation: The Micro and the Macro View*, pp. 71–102, World Scientific, 1992.

[8] M. Kocakulak and I. Butun, "An overview of Wireless Sensor Networks towards internet of things," in *Proceedings of the Computing and Communication Workshop and Conference (CCWC)*, pp. 1–6, IEEE, 2017, 7th Annual.

[9] J. Zheng and A. Jamalipour, *Wireless Sensor Networks: A Networking Perspective*, John Wiley & Sons, 2009.

[10] IETF, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Network," 2018. https://tools.ietf.org/html/rfc6550.

[11] T. IETF, "Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)," 2018. https://tools.ietf.org/html/rfc6552.

[12] G. Nikolic, T. Nikolic, M. Stojcev, B. Petrovic, and G. Jovanovic, "Battery capacity estimation of wireless sensor node," in *Proceedings of the IEEE 30th International Conference on Microelectronics (MIEL)*, pp. 279–282, IEEE, 2017.

[13] K. Kato and W. M. Bart, *Encyclopedia of Research Design*, SAGE Publications, Inc, 2012.

[14] T. Wang, Q. Wu, S. Wen et al., "Propagation modeling and defending of a mobile sensor worm in wireless sensor and actuator networks," *Sensors*, vol. 17, no. 1, p. 139, 2017.

[15] The R Project, The R Project Statistical Computing, 2018. https://www.r-project.org/.

[16] CRAN, "deSolve: Solvers for Initial Value Problems of Differential Equations," 2018. https://cran.r-project.org/web/packages/deSolve/index.html.

[17] CRAN, "MonteCarlo: Automatic Parallelised Monte Carlo Simulations," 2018. https://cran.r-project.org/web/packages/MonteCarlo/index.html.

[18] S. Karsten, G. Rave, and J. Krieter, "Monte Carlo simulation of classical swine fever epidemics and control: I. General concepts and description of the model," *Veterinary Microbiology*, vol. 108, no. 3-4, pp. 187–198, 2005.

[19] Y. Ji, L. Yao, S. Liu, H. Yao, Q. Ye, and R. Wang, "The study on the botnet and its prevention policies in the internet of things," in *Proceedings of the 2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pp. 837–842, Nanjing, China, May 2018.

[20] A. Singh, A. K. Awasthi, K. Singh, and P. K. Srivastava, "Modeling and analysis of worm propagation in wireless sensor networks," *Wireless Personal Communications*, vol. 98, no. 3, pp. 2535–2551, 2018.

[21] M. T. Gardner, C. Beard, and D. Medhi, "Using SEIRS Epidemic Models for IoT Botnets Attacks," in *Proceedings of the DRCN 2017 - Design of Reliable Communication Networks 13th International Conference*, pp. 1–8, 2017, Proceedings of VDE.

[22] B. K. Mishra and N. Keshri, "Mathematical model on the transmission of worms in wireless sensor network," *Applied Mathematical Modelling*, vol. 37, no. 6, pp. 4103–4111, 2013.

[23] L. Feng, L. Song, Q. Zhao, and H. Wang, "Modeling and stability analysis of worm propagation in wireless sensor network," *Mathematical Problems in Engineering*, vol. 2015, Article ID 129598, 8 pages, 2015.

[24] J. A. Jerkins and J. Stupiansky, "Mitigating IoT insecurity with inoculation epidemics," in *Proceedings of the ACMSE 2018 Conference*, p. 4, ACM, 2018.